**Software Engineering Assignment**

**MODULE: 5 (Database)**

**Basics of Database**

1. What do you understand by Database?

Ans. Database is a collection of information or data. It is possible to store, retrieve and edit this data by software called DBMS (**D**ata**b**ase **M**anagement **S**ystem).

1. What is Normalization?

Ans. Normalization is the process of minimizing redundancy (duplicity) from the database table. It divides larger table into smaller and links them using relationships. It works through series of stages called Normal Forms. Following are the various types of Normal Forms:

1. **First Normal Form (1NF):** It helps to eliminate duplicate data and simply queries.
2. **Second Normal Form (2NF):** It divides data into subgroups that exist in multiple rows of database table and represents it in a new table with connections made between them.
3. **Third Normal Form (3NF):** The relation that is in 1NF and 2NF, where none of the non-primary key attributes transitively depend on their primary keys.
4. **BCNF (Boyce Codd Normal Form):** It is an advanced version of 3NF.
5. What is difference between DBMS and RDBMS?

|  |  |  |
| --- | --- | --- |
|  | **DBMS** | **RDBMS** |
| 1. | DBMS stands for Database Management System | RDBMS stands for Relational Database Management System |
| 2. | Data is stored in form of files | Data is stored in form of tables |
| 3. | There is no existence of relationship between data | The data stored is related to each other |
| 4. | Normalization is not present | Normalization is present |
| 5. | It deals with small quantity of data | It stores large amount of data |
| 6. | Data redundancy is common in this model | It does not allow data redundancy |
| 7. | Security is less | More security measures are provided |
| 8. | It supports single user | It supports multiple user |

1. What is MF Codd Rule of RDBMS?

Ans. Dr. Edgar F. Codd (E.F Codd) was a Computer Scientist who, in 1985, invented the Relational model for Database management(RDBMS). He proposed 13(numbered 0-12) rules popularly known as Codd's 12 rules that actually defines what quality a DBMS requires in order to become a Relational Database Management System(RDBMS).

He stated the following rules:

Rule 0: The Foundation Rule

Rule 1: Information Rule

Rule 2: Guaranteed Access Rule

Rule 3: Systematic Treatment of Null Values

Rule 4: Active/Dynamic Online Catalog based on the relational model

Rule 5: Comprehensive Data SubLanguage Rule

Rule 6: View Updating Rule

Rule 7: Relational Level Operation (High-Level Insert, Update and Delete) Rule

Rule 8: Physical Data Independence Rule

Rule 9: Logical Data Independence Rule

Rule 10: Integrity Independence Rule

Rule 11: Distribution Independence Rule

Rule 12: Non Subversion Rule

1. What do you understand by Data Redundancy?

Ans. Data Redundancy means when multiple copies of same information (data) is stored in more than one place at a time.

1. What is DDL Interpreter?

Ans. It interprets the DDL (Data Definition Language) instructions (tabular format) and stores the record in data dictionary/metadata (data of the database).

1. What is DML compiler in SQL?

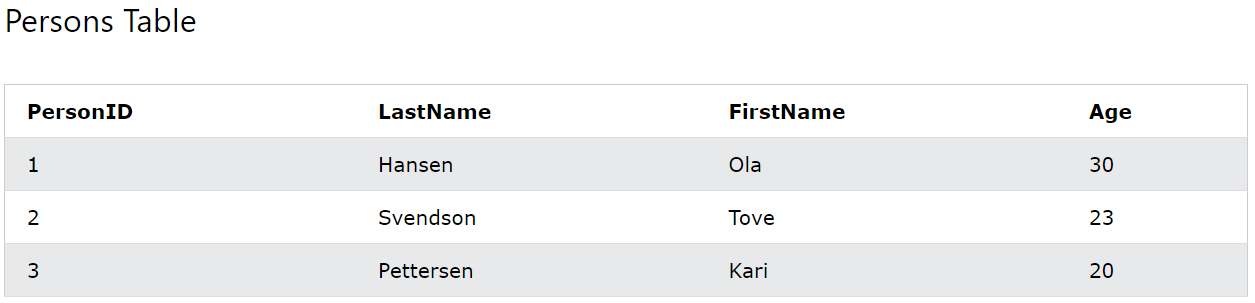
Ans. It converts DML (Data Manipulation Language) instructions into machine language (low level language).

1. What is SQL Key Constraints writing an example of SQL key constraints?

Ans. SQL Key Constraints are used to specify rules for the data in a table. It limits the type of data that goes into the table to ensure the accuracy and reliability of the data in the table. Any violation to this will abort the action.

The following constraints are commonly used in SQL:

1. **Primary Key**: It is a combination of not null (never blank) and unique data.

Example:

CREATE TABLE Persons (

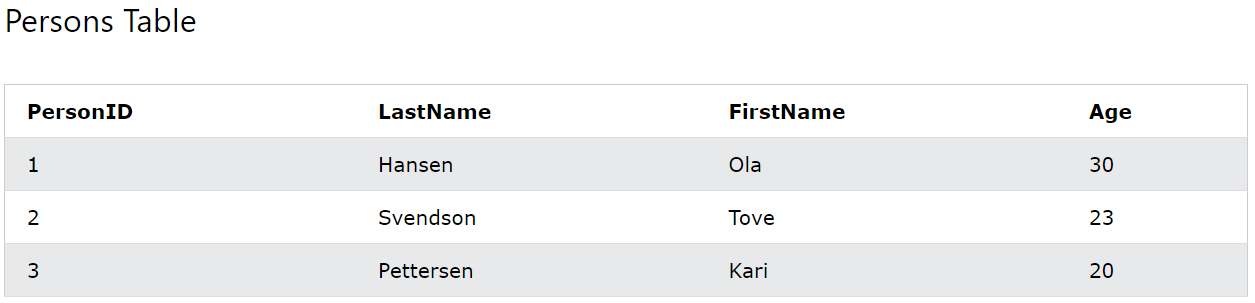
**ID int NOT NULL PRIMARY KEY,**

LastName varchar(50) NOT NULL,

FirstName varchar(50),

Age int );

1. **Unique Key**: It ensures that data is not repeated (values in every column is different) and can be null.

Example:

CREATE TABLE Persons (

**ID int NOT NULL UNIQUE,**

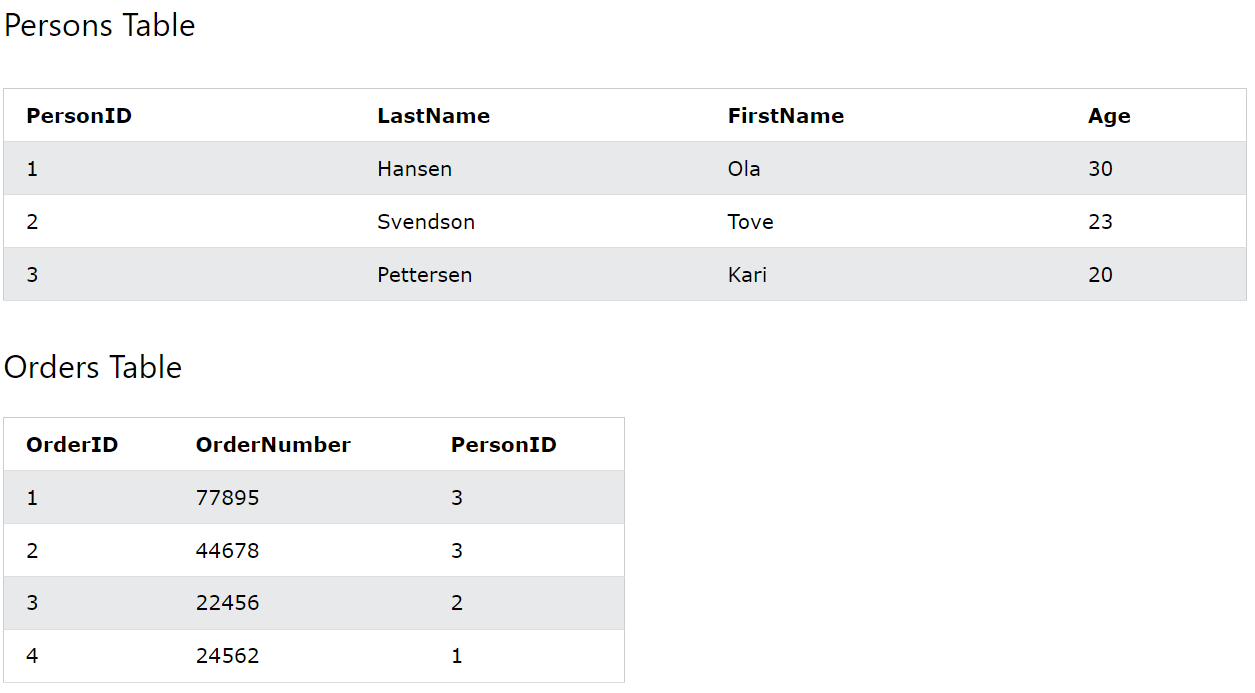
LastName varchar(50) NOT NULL,

FirstName varchar(50),

Age int );

1. **Foreign Key:** It is used to maintain a relation between two tables.

Example:



CREATE TABLE Persons (

**ID int NOT NULL PRIMARY KEY,**

LastName varchar(50) NOT NULL,

FirstName varchar(50),

Age int

);

CREATE TABLE Orders (

OrderID int NOT NULL,

OrderNumber int NOT NULL,

PersonID int,

PRIMARY KEY (OrderID),

**FOREIGN KEY (PersonID) REFERENCES Persons(PersonID)**

);

1. What is Save Point? How to create a save point? Write a query?

Ans. Save Point is a command in SQL to roll back to a certain point without rolling back the entire table.

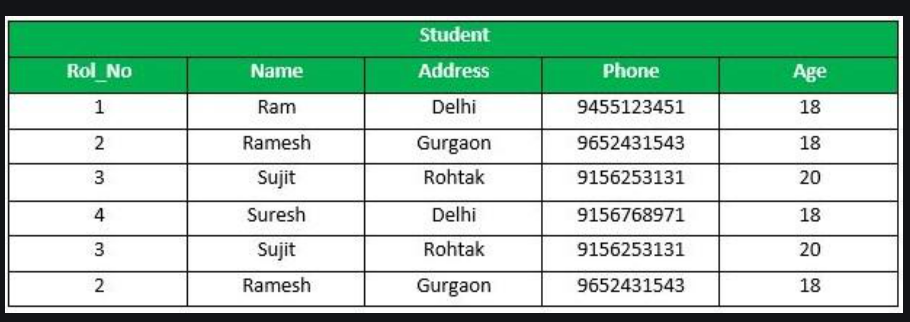
* Syntax for Savepoint command:

SAVEPOINT SAVEPOINT\_NAME;

* Syntax for rolling back to Savepoint command:

ROLLBACK TO SAVEPOINT\_NAME;

* **Query Example:**

*From the above example Sample table1, Delete those records from the table which have age = 20 and then ROLLBACK the changes in the database by keeping Savepoints.*

**Query:**

SAVEPOINT SP1; //Savepoint created

DELETE FROM Student WHERE Age=20; //deleted

SAVEPOINT SP2; //Savepoint created

**Output:**

1. What is trigger and how to create a trigger in SQL?

Ans. Trigger is used to store history of the new data inserted/updated/deleted in predefined database. It automatically updates a record in one table whenever a record is inserted into another table.

Syntax to create a trigger:

DELIMITER $$

CREATE TRIGGER `TRIGGER\_NAME` (AFTER | BEFORE) {INSERT | UPDATE | DELETE} ON `table\_name` FOR EACH ROW

BEGIN

INSERT into `table\_name` SET `column\_name = new.column\_name | old.column\_name;

END$$

What is database and why do we use it?

Database is a collection of information or data. It is possible to store, retrieve and edit this data by software called DBMS (**D**ata**b**ase **M**anagement **S**ystem). It facilitates efficient storage, retrieval, and management of information.

What is DBMS? What operations can be performed in DBMS?

DBMS stands for Database Management System. It is software that provides an interface for interacting with databases and facilitates the creation, maintenance, and use of databases. DBMS acts as an intermediary between the users and the database, ensuring efficient and secure management of data.

There are several operations that can be performed in a DBMS:

**Data Definition Language (DDL) Operations:**

Create: Used to create a new database, table, index, or other database objects.

Alter: Modifies the structure of an existing database object, such as adding or deleting columns in a table.

Drop: Deletes an existing database, table, index, or other database objects.

**Data Manipulation Language (DML) Operations:**

Select: Retrieves data from one or more tables based on specified criteria.

Insert: Adds new records or rows into a table.

Update: Modifies existing records in a table based on specified conditions.

Delete: Removes records from a table based on specified conditions.

**Data Query Language (DQL) Operations:**

SELECT: Used to query the database and retrieve data. This is a subset of DML and is primarily concerned with data retrieval.

**Transaction Control Operations:**

Commit: Confirms the changes made during a transaction and makes them permanent.

Rollback: Undoes the changes made during a transaction, reverting to the state before the transaction started.

Savepoint: Establishes points within a transaction to which you can later roll back.

**Data Control Language (DCL) Operations:**

GRANT: Provides specific privileges to users or roles.

REVOKE: Removes specific privileges from users or roles.

**Data Integrity Operations:**

Constraints: Define rules to maintain data integrity, such as unique constraints, primary key constraints, foreign key constraints, etc.

Triggers: Special stored procedures that are automatically executed in response to certain events, ensuring data consistency and integrity.

**Security Operations:**

User Management: Involves creating, modifying, and deleting user accounts and managing their access privileges.

Authentication and Authorization: Ensures that only authorized users can access the database and perform specific operations.

**Concurrency Control Operations:**

Locking Mechanisms: Prevents conflicts when multiple users attempt to access or modify the same data simultaneously.

These operations collectively enable users to define, manipulate, query, and control the data stored in a database, ensuring data integrity, security, and efficient management within the DBMS environment.

What is RDBMS?

RDBMS stands for Relational Database Management System. It is a type of database management system that is based on the relational model of data. In the relational model, data is organized into tables, where each table consists of rows and columns. Each row represents a record, and each column represents an attribute or field of the record.

Key characteristics of RDBMS include:

**Tables:** Data is organized into tables, which are two-dimensional structures with rows and columns. Each table has a unique name and consists of records (rows) and attributes (columns).

**Relationships:** RDBMS supports relationships between tables through the use of keys. A primary key uniquely identifies each record in a table, and foreign keys establish links between tables.

**Data Integrity:** RDBMS enforces data integrity through the use of constraints such as primary key constraints, foreign key constraints, unique constraints, and check constraints. These constraints help maintain the accuracy and consistency of data.

**ACID Properties:** RDBMS ensures the ACID properties of transactions - Atomicity, Consistency, Isolation, and Durability. This ensures that database transactions are reliable and maintain the integrity of the data.

**Structured Query Language (SQL):** RDBMS uses SQL as the standard language for querying and manipulating the data. SQL allows users to perform operations like SELECT, INSERT, UPDATE, and DELETE on the data in the database.

**Normalization:** RDBMS supports the normalization process to eliminate data redundancy and improve data integrity. Normalization involves organizing the database schema to reduce data duplication and dependency.

**Concurrency Control:** RDBMS includes mechanisms for managing concurrent access to the database by multiple users, ensuring that transactions do not interfere with each other.

Popular examples of RDBMS include:

MySQL: An open-source relational database management system.

Oracle Database: A comprehensive and widely used commercial RDBMS.

Microsoft SQL Server: A relational database management system developed by Microsoft.

PostgreSQL: An open-source object-relational database system.

RDBMS is widely used in various applications and industries due to its flexibility, scalability, and the ability to model complex relationships between data. It serves as a foundation for many modern information systems, supporting the storage and retrieval of structured data in a relational format.

Explain subsets of DB.

What is SQL?

SQL, which stands for Structured Query Language, is a standard programming language specifically designed for managing and manipulating relational databases. It is used to interact with relational database management systems (RDBMS) and is widely employed for tasks such as querying data, updating data, inserting data, and managing the structure of a database.

Key characteristics of SQL include:

**Data Query Language (DQL):**

SELECT: Used to retrieve data from one or more tables in a database.

**Data Definition Language (DDL):**

CREATE: Creates a new database object, such as a table, view, or index.

ALTER: Modifies the structure of an existing database object, like adding or dropping columns in a table.

DROP: Deletes an existing database object.

**Data Manipulation Language (DML):**

INSERT: Adds new records or rows into a table.

UPDATE: Modifies existing records in a table based on specified conditions.

DELETE: Removes records from a table based on specified conditions.

**Data Control Language (DCL):**

GRANT: Provides specific privileges to users or roles.

REVOKE: Removes specific privileges from users or roles.

**Transaction Control Language (TCL):**

COMMIT: Confirms the changes made during a transaction and makes them permanent.

ROLLBACK: Undoes the changes made during a transaction, reverting to the state before the transaction started.

SQL is used by developers, database administrators, and analysts to interact with databases, retrieve information, and perform various operations on the data. It provides a standardized way to communicate with different relational database management systems, including popular ones like MySQL, PostgreSQL, Microsoft SQL Server, Oracle Database, and SQLite.

SQL commands are written in a declarative style, meaning that users specify what they want the database to do, and the database management system figures out how to execute the request. The language is essential for anyone working with databases and is a fundamental skill for individuals involved in software development, data analysis, and database administration.

Difference between SQL and MySQL.

SQL and MySQL are related but distinct entities:

**SQL:**

Definition: SQL, or Structured Query Language, is a standard programming language used for managing and manipulating relational databases.

Purpose: SQL is a language that provides a set of commands for interacting with relational database management systems (RDBMS). It includes commands for querying data, updating data, inserting data, and managing the structure of a database.

Usage: SQL is a language used universally for relational databases. It is not tied to any specific database system but is a standard that multiple database systems adhere to.

**MySQL:**

Definition: MySQL is a specific relational database management system (RDBMS) that uses SQL as its query language.

Purpose: MySQL is designed to store, manage, and retrieve data in a relational database format. It supports SQL for interacting with the database, and it also includes additional features specific to the MySQL system.

Ownership: MySQL is an open-source RDBMS, and it is owned by Oracle Corporation. However, it is available under an open-source license, and there are community editions as well as commercial versions with additional features and support.

In summary, SQL is a standardized language for working with relational databases, while MySQL is a specific implementation of an RDBMS that uses SQL as its query language. MySQL is just one of many relational database systems that support SQL. Other examples include PostgreSQL, Microsoft SQL Server, Oracle Database, and SQLite. When people refer to "SQL," they are usually talking about the language itself, whereas "MySQL" specifically refers to the MySQL database management system.

What are constraints in SQL?

In SQL, constraints are rules or conditions applied to the columns or tables to enforce data integrity and ensure that the data stored in the database meets certain criteria. Constraints define limits or restrictions on the type of data that can be inserted, updated, or deleted in a table. They help maintain the accuracy, consistency, and reliability of the data within the database. Here are some common types of constraints in SQL:

**Primary Key Constraint:**

Purpose: Ensures that each record in a table is uniquely identified by a specific column or set of columns. It is used to prevent duplicate and null values in the primary key column(s).

Syntax:

CREATE TABLE table\_name (

column1 datatype PRIMARY KEY,

column2 datatype,

...

);

**Unique Constraint:**

Purpose: Ensures that each value in a specified column or set of columns is unique across all records in the table.

Syntax:

CREATE TABLE table\_name (

column1 datatype UNIQUE,

column2 datatype,

...

);

**Check Constraint:**

Purpose: Specifies a condition that must be satisfied for each row in a table. It allows the definition of custom rules to restrict the range of allowable values.

Syntax:

CREATE TABLE table\_name (

column1 datatype,

column2 datatype,

...

CHECK (condition)

);

**Foreign Key Constraint:**

Purpose: Establishes a link between two tables by enforcing referential integrity. It ensures that values in a column (or columns) of one table correspond to the values in a referenced column (usually the primary key) of another table.

Syntax:

CREATE TABLE table\_name1 (

column1 datatype PRIMARY KEY,

column2 datatype,

...

);

CREATE TABLE table\_name2 (

column1 datatype,

column2 datatype,

...

FOREIGN KEY (column1) REFERENCES table\_name1(column1)

);

**Not Null Constraint:**

Purpose: Ensures that a column does not contain NULL values.

Syntax:

CREATE TABLE table\_name (

column1 datatype NOT NULL,

column2 datatype,

...

);

What is meaning of primary key and how many primary key can a table have?

The primary key is crucial for identifying and establishing relationships between tables in a relational database. It is often used as a reference in foreign keys of other tables, creating links between related data.

As for the question of how many primary keys a table can have, the answer is that a table can have only one primary key. However, a primary key can consist of a single column or a combination of multiple columns. In the case of a composite primary key (using multiple columns), the combination of values in those columns must be unique across all records in the table. This uniqueness constraint is collectively enforced for the entire set of columns comprising the primary key.

We have below field data: ID, student name, email, contact number, city… which constraints can be used?

CREATE TABLE students (

ID INT PRIMARY KEY NOT NULL,

student\_name VARCHAR(255) NOT NULL,

email VARCHAR(255) UNIQUE NOT NULL,

contact\_number VARCHAR(15) CHECK (LENGTH(contact\_number) >= 10),

city VARCHAR(100)

);

What is meaning of Foreign key and how many FK can we have in one table?

A foreign key (FK) is a column or a set of columns in a relational database table that refers to the primary key or a unique key in another table. It establishes a relationship between the tables based on the values stored in the foreign key columns and the referenced primary key or unique key columns. The purpose of a foreign key is to maintain referential integrity between related tables.

In general, it is common to have multiple foreign keys in a table, especially when dealing with complex relationships between tables. Each foreign key in a table represents a different relationship with other tables in the database. It is essential to design foreign keys carefully to maintain data integrity and ensure that relationships between tables are properly enforced.

Difference between truncate, drop and delete.

Use TRUNCATE when you want to quickly remove all rows from a table without logging individual deletions and don't need the operation to be rolled back.

Use DROP when you want to completely remove a table and all its associated structures from the database (irreversible).

Use DELETE when you need to selectively remove specific rows from a table and require the option to roll back the changes within a transaction.